Come with Me Now: New Potential Consumers Identification from Competitors*

Hugo Alatrista-Salas[1][0000−0001−5252−4728], Miguel Nunez-del-Prado[1][0000−0001−7997−1739], and Victoria Zevallos[1]

[1]Universidad del Pacífico, Lima - Peru
{h.alatristas,m.nunezdelpradoc,v.zevallosmunguia}@up.edu.pe

Proceedings version

Abstract. The telecommunications industry is confronted more and more to aggressive marketing campaigns from competitor carriers. Therefore, they need to improve the subscriber targeting to propose more attractive offers for gaining new subscribers. In the present effort, a five steps methodology to find new potential subscribers using supervised learning techniques over imbalanced datasets is proposed. The proposed technique applies community detection to infers consumption information of competitors carriers subscribers within the communities. Besides, it uses a sampling technique to reduce the effect of a dominant class for an imbalanced classification task. The proposal is evaluated with a real dataset from a Peruvian carrier. The dataset contains one-month data, which is about 200 millions of transaction. The results show that the proposed technique is able to identify between two to ten times more new potential clients, depending on the sampling technique, as shows using the top decile lift value.

Keywords: Subscribers attraction · imbalanced classification · community detection.

1 Introduction

Competition among telecom operator has radically increased in recent years in Latin America. As a result, operators are making significant investments in developing new strategies allowing them to increase their market share. These strategies have three different approaches. The first is to attract new users to the industry; the second, to attract customers of the competitors; and the third, in retaining the clients. While all fronts are important, the objective of the following study is to attract new clients from other telecom operators. This task presents considerable and interesting challenges due to the lack of information about the subscriber behavior from other telecom operators. Therefore, we rely on the information of the interaction these users maintain with the company's subscribers to determine their future behavior. The main objective is determining which clients belonging to another telecom operator are more likely to

* Authors appear in alphabetical order, they contribute equally to the present paper.

become new subscribers based on the analysis performed on the interactions in the telecommunications network.

To attain this objective, we use Call Detail Records (*i.e.,* call traffic and internet consumption) of Small Office Home Office and post-pay subscribers to determine whether a subscriber from another telecom operator will become a subscriber. We model the structure of the mobile social network as a directed graphs G(V,E), where the vertices are weighted by the internet consumption and edges by the volume of incoming and outgoing calls. Then, communities are detected to infer information about the subscribers competitors behavior based on the attributes of the subscribers sharing the same community. Once information is completed, we compute some variables to qualify the changes in subscribers attributes over time. Finally, a classification algorithm is applied to identify the most likely subscribers to migrate to the carrier running our methodology. Our approach achieves an accuracy value around 0.9. However, this classification is not trivial since the dataset is imbalanced. Therefore, we also performed a comparative analysis of resampling techniques to balance the dataset before performing the classification task.

The present work is organized as follows. Section 2 presents the related works. Section 3 introduces our methodology, while Section 4 describes the dataset we have used. Section 5 details the result of our experiments. Finally, Section 6 presents the conclusions and new research avenues.

2 Related Works

In this section, we list different studies on churn prediction, which is basically the same prediction task we perform. For instance, Columelli, Nunez-del-Prado and Zarate [2] introduce a methodology that summarizes churn risk score in telecommunication social networks. They rely on Fuzzy Logic system, combining the churn probability and the risk of the churner to leave the network with other subscribers. Their objective is classifying the possible deserters and calculate their influence analyzing the social network of an African telecommunication operator. First, they make a comparison between several classifiers, where the Extremely Random Tree algorithm obtains a better performance according to the lift curve. Then, they use metrics such as degree of centrality and page rank to measure the degree of influence issued and received from each possible churner. Finally, they apply a fuzzy logic system to obtain a unified metric of churn risk based on the measures of the probability of desertion, influence emitted (degree of centrality) and influence received (page rank) [2]. This paper has value because it proposes an interesting methodology to measure the risk of churn and makes a comparison between classification algorithms to predict the behavior of the users.

Pushpa and Shobha [10] propose to analyze the structure and behavior of a multi-relational network and the location of important elements to predict the churn of customers of a mobile operator in India. With this purpose, the social position of the users, represented by nodes, is evaluated based on the multiple

Proof Preparation

connections they have with other users of the network (centrality degree). Then, this value is used to characterize the degrees of influence and importance of certain members. Finally, the REGE iterative algorithm based on regular equivalence (similarity between relationships) is used for classified users as deserters or non-deserters [10]. The value of this research relies on the analysis of the structure of the telecommunication network and the social importance of the nodes. This knowledge is then used to predict the churn, but it can also be used to attract customers more likely to leave the network of a competitor operator.

Amin *et al.* [1] propose a just in time classification technique using Naive Bayes. The idea behind this work is to use another company data to train a churn classifier for another company. Authors also test whether data transformation improves the precision of the prediction. They test their approach with publicly available data. The first dataset contains 20 features and 333 samples, and the second dataset has 250 features and 18000 customers. The metric they use to evaluate the performance of the Naive Bayes classification is the Area Under the ROC curve.

De Caigny, Coussement, and De Bock [3] compare Decision Tree, Logistic Regression, Random Forests and Logit Leaf Models for churn prediction. The idea behind LLM is that models constructed on different segments of the data rather than on the entire dataset predict better while maintaining the model interpretability. The LLM consists of two steps. In the first step, subscribers are segmented using decision rules. In the second step, a model is created for every leaf of the tree. The area under the receiver operating characteristics curve (AUC) and top decile lift (TDL) are used to measure the performance for which LLM scores significantly better than Logistic Regression and Decision Trees. It also performs as well as Random Forests and Logistic model trees. To perform the evaluations, the authors use 14 datasets from different industries of the Center for Customer Relationship Management Duke University.

3 Methodology

In the present section, we describe the methodology to attract new costumers from competitors in the telecommunications industry. The present effort allows telecommunication operators to determine which competitor operators subscribers are more likely to become new subscribers based on five different steps, as depicted in Figure 1. Namely, graph modeling, community detection, variable generation, balanced classification, and feature analysis.

First, the *graph modeling* phase consist on building the social graph from the Call detail Records (*CDR*). The nodes stand for both *own subscriber* from the carrier running the analysis and subscribers from competitor carriers, while the edges represent social relations [10]. Thus, analyzing the social network provides essential information about the interaction of *own subscribers* and subscribers from competitors.

The *community detection* phase extracts the subgroup of the social network represented by a graph [10]; where the density of the connections is high within

Preprint version

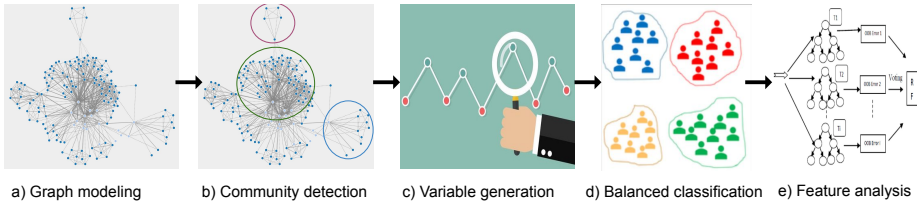


Fig. 1. Methodology process.

the community; while, the links between the nodes of different communities tend to be of low density [8]. The extraction of relevant communities in large social networks is a real challenge; however, it provides important information for making decisions about the structure and behavior of the subscribers in graph and subgraphs [11]. Therefore, we apply community detection to understand the competitors' subscribers behavior based on the characteristics of the *own subscribers* who belong to the same community. To accomplish this task, we rely on an adaptation of the Louvain algorithm. This variant consists on the change of the conventional modularity formula to work with directed graphs. This adjustment was proposed by Dugué and Perez *et al.* [4] based on Leicht and Newman work [6]. Hence, this algorithm allows to maintain the directionality and uses the computational effectiveness of Louvain algorithm to work with high dimensional networks.

$$Q_d = \frac{1}{m} \sum_{i,j} [A_{i,j} - \frac{d_i d_j}{2m}] \delta(c_i, c_j) \tag{1}$$

Where m is the number of edges in G, A_{ij} is the weight of the edge between nodes i and j, which is set to 0 if such edge does not exist, d_i is the number of neighbors of i (i.e. the degree of vertex of i), c_i is the community to which vertex i belongs and the $\delta(c_i, c_j)$ is defined as 1 if $c_i = c_j$, and 0 otherwise.

In the third phase of our methodology, we generate new variables to characterize competitors subscribers, Accordingly, based on the social graph from the CDR, we build features like out-degree, in-degree, centrality out-degree, centrality in-degree and page rank. Also, we compute measures per community such as percentage of *own subscribers*, average number of consumed megabytes, average number of internet access, out-degree normalized, in-degree normalized, centrality out-degree normalized, and centrality in-degree normalized. In addition, we calculate ratios of the variables to quantify variation between weeks t and $t-1$ as part of the characterization. These ratios are the variation of number of megabytes, number of internet access, out-degree normalized, in-degree normalized, centrality out-degree normalized, and centrality in-degree normalized. All these new variables are computed for each week to predict whether a competitor's subscriber could leave competitor carriers to become a client of the telecom operator running this proposal.

Processing version

In our case, as the dataset is imbalanced. In the fourth phase we compare three different families of resampling techniques to avoid bias in the classification results [7]. Thus, we compare five techniques of *over-sampling* (*i.e.,* Naive random over-sampling, Synthetic Minority Oversampling Technique (SMOTE), Borderline-1 SMOTE, Borderline-2 SMOTE and Adaptive Synthetic (ADASYN)), seven techniques of *under-sampling* (*i.e.,* Random under-sampling, NearMiss-1, NearMiss-3, Edited Nearest Neighbor, All KNN, One Side Selection and Neighborhood Cleaning Rule), and a hybrid technique of *over-sampling and under-sampling* (*i.e.,* SMOTE Tomek).

The last phase uses the aforementioned variables and balancing techniques for identifying competitors subscribers who will change their carrier from those who would not leave their current carrier in a certain period of time. With this objective, we use a binary classifier which uses the data of the first three weeks to train and the data of the last week to test the effectiveness of the model. To choose the classification algorithm, we rely on work of Columelli *et al.* [2]. Authors realized a comparative analysis between different classification algorithms, such as Extremely Random Trees, Naive Bayesian and Gradient Boosting, with the objective of predicting prepaid mobile subscribers desertion. The results show that the best classifier was Extremely Random Trees.

In the next section, we describe the dataset at our disposal to perform experiments.

4 Dataset Description

For the experiments, we use telecommunication anonymized data derived from a CDR of a Peruvian telecommunication operator. The data provides 121 310 940 call events and 106 400 759 internet connections of four weeks during June of 2018. It is worth noting that subscribers are tagged as *portability i.e.,* subscribers that came from a competitor carrier. From this data, we have split five different and complementary datasets as described below:

- **Call traffic data** is composed of Caller Id (*i.e.,* Subscriber Id), Callee Id, direction (incoming, outgoing), date, hour, duration in minutes, number of exchanged calls, average call duration, and type (On net, calls between subscribers of the same carrier; Off net, calls between subscriber and competitors subscriber). We present an example of this dataset in Table 1.

- **Internet consumption data** contains only information about the subscribers but not from subscribers belonging to other carriers. It comprises Subscriber ID, date, hour, and number of consumed megabytes.

- **Portability data** identifies which subscribers came from a competitor operator. This variable is used to predict whether a client will change carrier from a competitor telecom operator.

Pre-reading version

Caller Id	Callee Id	Direction	date	Hour	Min	Calls	Avg	Type
948000000	948819472	IN	01/06/2018	18:32:29	394	9	43.77	ON-NET
948000000	950171864	OUT	01/06/2018	01-13:19	45	1	22	ON-NET
948011126	950171864	IN	02/06/2018	11:12:35	3	1	3	OFF-NET
948000000	951557467	OUT	04/06/2018	07:52:29	46	1	46	OFF-NET
948000000	951897248	IN	11/06/2018	12:03:09	8	1	8	OFF-NET

Table 1. Call traffic data example. Where *min* is the number of minutes of all calls; *Calls* is the Number of Calls; *Avg* is the average minutes per call

Subscriber Id	Date	Hour	Megabytes
948000000	11/06/2018	07:52:21	96
948000001	17/06/2018	10:23:09	85
948819472	19/06/2018	17:35:13	152
948000003	21/06/2018	21:12:51	157
950171864	30/06/2018	02:02:37	96

Table 2. Internet consumption data example. *Megabytes* is the consumed number of Megabytes,

Subscriber Id	Portability
948000000	0
948000001	0
948819472	1
948000003	1
950171864	0

Table 3. Portability data example. Where *Portability* refers to subscriber who came from other carriers,

- **Derived data from CDR**. Based in the first two data sets *i.e.,* tables 1 and 2. we transform and summarize the variables in four weeks. Having as a result a dataset per week. Table 4.

- **Social network data** is modeled as a graph $G(N, E)$ where the nodes are the subscribers and edges represents calls between subscribers. This information is extracted from Table 1. Thus, We construct three directed graphs weighted by the total number of minutes of calls, total number of calls, and average minutes per call. It is worth noting that the these graph are generated per week.

In the next section, we describe the the results of our experiments based on the described methodology and datasets.

5 Experiments

In the present section, we present the results of our experiments. First, we build the graph models weighting the edges by the number of calls, the amount of

Proofreading Version

Variable	Description
out_x	Out degree
in_x	In degree
cent_out_x	Centrality out degree
cent_in_x	Centrality in degree
pagerank_x	Page rank
%_own_subscribers_x	Percentage of own subscribers (*i.e*) in the community
megas_prom_x	Average number of megabytes used by own subscribers in the community
use_prom_x	Average number of times of internet use by own subscribers in the community
out_norm_x	Out degree normalized based on the community
in_norm_x	In degree normalized based on the community
cent_out_norm_x	Centrality out degree normalized based on the community
cent_in_norm_x	Centrality in degree normalized based on the community
ratio_megas_prom_x	Ratio between megas_prom_x of the week "t" and the megas_prom_x of the week "t-1"
ratio_use_prom_x	Ratio between use_prom_x of the week "t" and the use_prom_x of the week "t-1"
ratio_out_norm_x	Ratio between out_norm_x of the week "t" and the out_norm_x of the week "t-1"
ratio_in_norm_x	Ratio between in_norm_x of the week "t" and the in_norm_x of the week "t-1"
ratio_cent_out_norm_x	Ratio between cent_out_norm_x of the week "t" and the cent_out_norm_x of the week "t-1"
ratio_cent_in_norm_x	Ratio between cent_in_norm_x of the week "t" and the cent_in_norm_x of the week "t-1"

Table 4. Derived data from CDR description.

minutes and the average of minutes per calls for four weeks $S1$, $S2$, $S3$, and $S4$. Table 5 shows the size of the generated graphs.

week	Nodes	Edges
S1	4 662 846	9 114 606
S2	4 704 090	9 080 917
S3	4 494 577	8 712 714
S4	4 198 888	7 939 203

Table 5. Graphs generated per week

Using the twelve generated graphs, we extract different variables taking the graphs weighted by the number of calls, the amount of minutes, and the average of minutes per calls. After applying community detection algorithm in each graph, we derived variables as summaries Table 4. It is worth noting that the x suffix in Table 4 is changed for c, m, or a when the variable is issued from

Proceeding Version

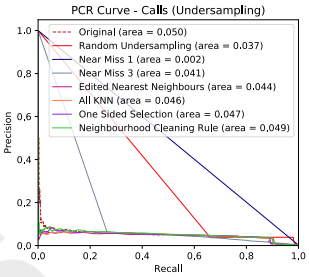
the graph of calls, minutes and average minutes per call, respectively. For instance, out_c, out_m, and out_a are the out_degree values for the calls, minutes and average graphs. Once we have obtained all the variables, we apply different techniques to balance the dataset for classification. Thus, to evaluate the performance of the different models and determinate which resampling technique and data sets is the most appropriate for the proposed task we use the metrics presented below.

To evaluate a classification with imbalanced data, a recommended measure to use is the precision / recall curve (PCR) [9]. It measures the trade off between precision and recall (sensitivity). As complement the average precision metric (AP) calculates the average with weights reached in each threshold, where the increase in recall compared to the previous threshold is used as a weight [12]. Figure 2 presents a comparison between the PCR curves of the resampling methods applied to the datasets (*i.e.,* total number of minutes, total number of calls, average minutes per call, and union of the three data sets). We observe that the silhouette and AP values, in general, do not show optimal results for classification. This scenario is explained by the low value obtained by the models in precision, as consequence of the imbalanced test data.

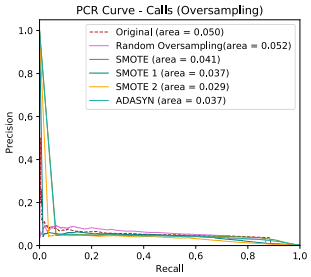
Another important performance visualization tool is ROC curve, which measures the trade off between specificity and sensitivity (recall) [5]. In addition, it provides a performance metric called AUC, which measures the area under the ROC curve. The AUC is useful for evaluating the performance of different classifiers [9]. Figure 3 presents a comparative between the techniques of under-sampling, over-sampling, and the combination of over-sampling and under-sampling applied to each of the data sets. We observe that the best Under-sampling techniques are Random under-sampling and NearMiss-3, while, the best Over-sampling techniques are SMOTE and ADASYN. Comparing the AUC values, we note that the one obtaining the best results for the datasets of calls, average and total is the Random under-sampling (AUC = 0.96) and the best for the minutes dataset is NearMiss-3 (AUC=0.97).

Another adequate metric used in previous works to evaluate churn models is the Lift curve and value [2]. Lift measures the effectiveness of a predictive model calculated as the ratio between the results obtained with and without the predictive model. Figure 4 compares the performance of the models in terms of Lift curve and the value at 10%. On the one hand, the comparison between under-sampling techniques, the ones with the best lift value at 10% are Random under-sampling and NearMiss-3 with 9.9 for all the data sets. On the other hand, in the case of over-sampling techniques, the best one is ADASYN with a lift value at 10% of 8.9, 9.0, 8.6 and 8.1 for calls, minutes, average, and total, respectively. Finally, in the case of the hybrid method SMOTE Tomek the lift value at 10% is 9.0, 8.6, 8.7, and 7.9 respectively. According to this results the best of all the models are Random under-sampling and NearMiss-3. Concerning the ground truth, we use the last week of the four weeks dataset. Therefore, we used the first three weeks to predict users who will come from other telecom operators

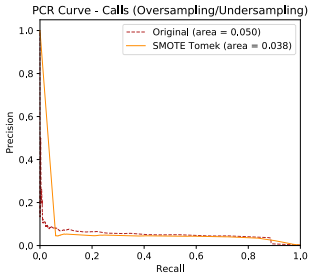
Preprint Version for Review



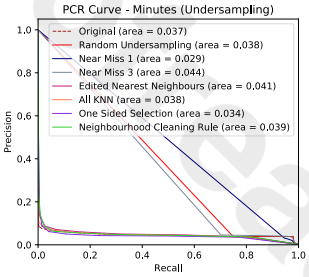
a) Calls undersampling



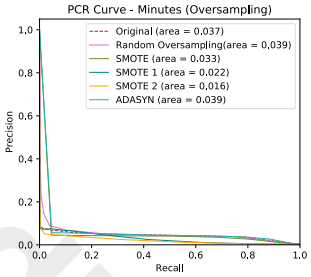
b) Calls oversampling



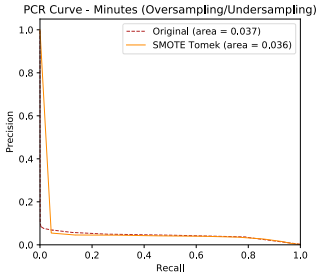
c) Calls oversampling and undersampling



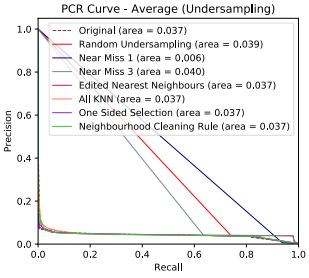
d) Minutes undersampling



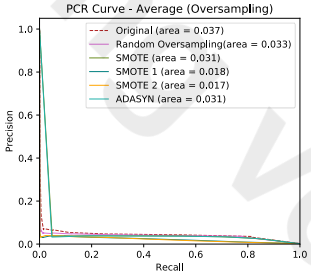
e) Minutes oversampling



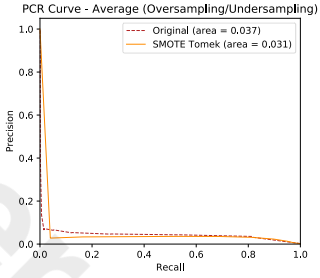
f) Minutes oversampling and undersampling



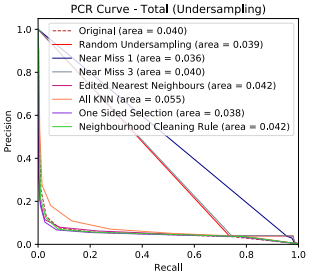
g) Average undersampling



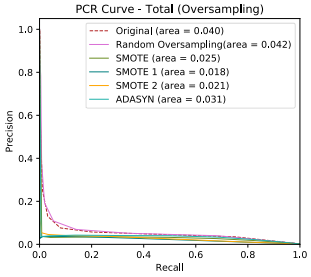
h) Average oversampling



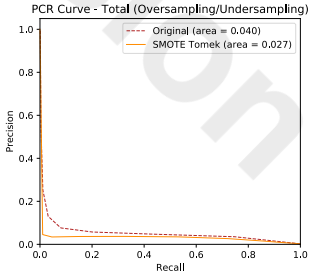
i) Average oversampling and undersampling



j) Total undersampling



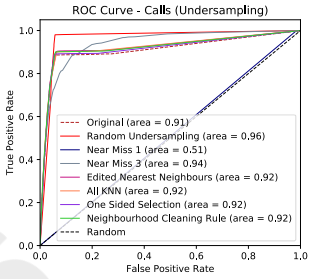
k) Total oversampling



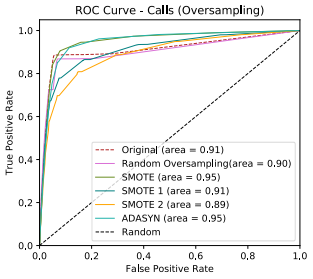
l) Total oversampling and undersampling

Fig. 2. Precision / recall curve (PCR) curve.

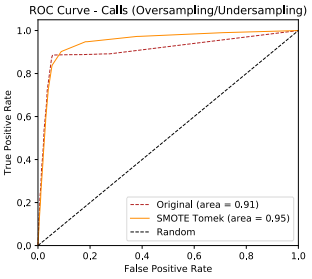
Proof reading version



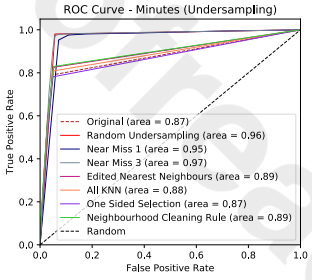
a) Calls undersampling



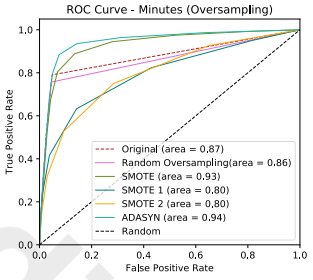
b) Calls oversampling



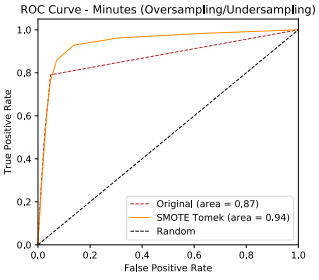
c) Calls oversampling and undersampling



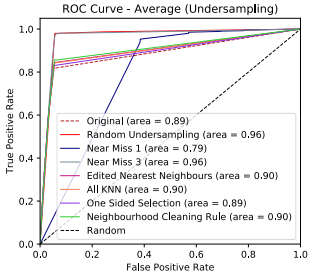
d) Minutes undersampling



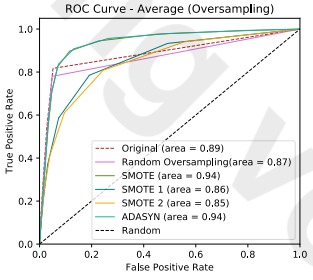
e) Minutes oversampling



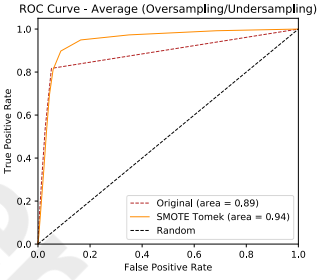
f) Minutes oversampling and undersampling



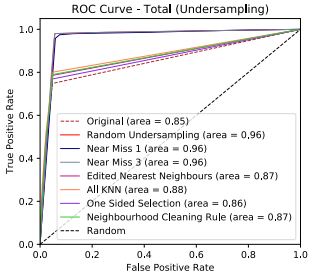
g) Average undersampling



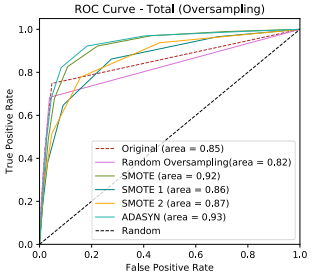
h) Average oversampling



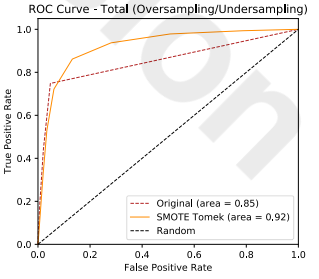
i) Average oversampling and undersampling



j) Total undersampling



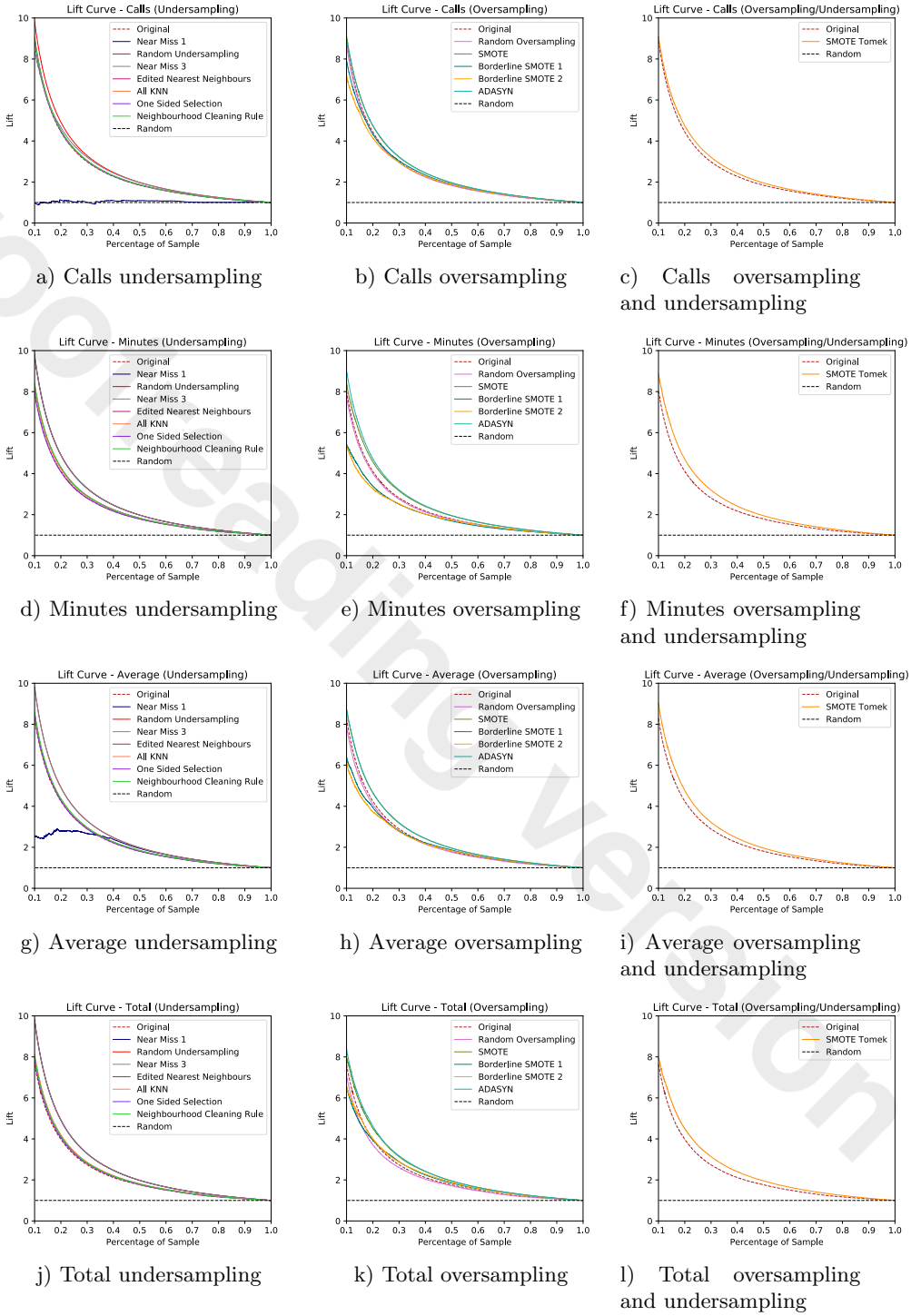
k) Total oversampling



l) Total oversampling and undersampling

Fig. 3. Receiver operating characteristic (ROC) curve.

Proof reading version



a) Calls undersampling b) Calls oversampling c) Calls oversampling and undersampling

d) Minutes undersampling e) Minutes oversampling f) Minutes oversampling and undersampling

g) Average undersampling h) Average oversampling i) Average oversampling and undersampling

j) Total undersampling k) Total oversampling l) Total oversampling and undersampling

Fig. 4. Lift curve.

Proofreading version

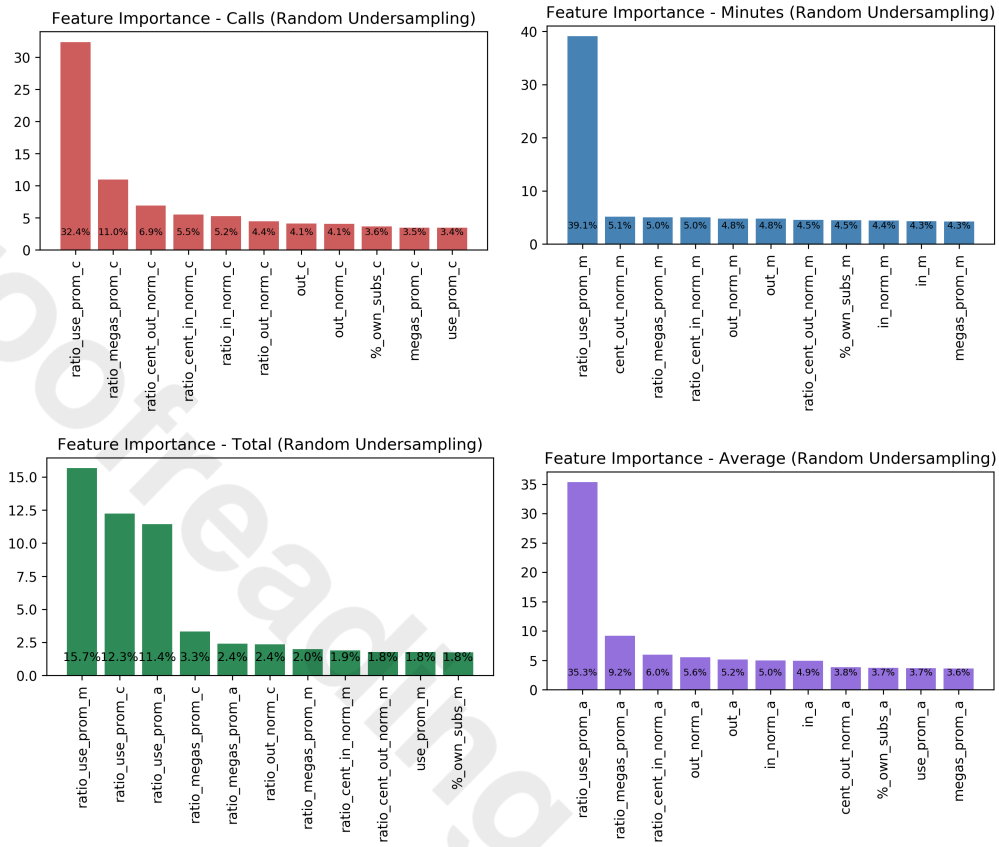


Fig. 5. Features importance analysis.

and we observe whether those predicted subscribers become own subscribers in the last week of the dataset.

Finally, Figure 5 depicts the reports of the importance of estimators of the four best classification tasks applying Random under-sampling and the Extremely Random Trees classifier. We see that the most important characteristics for all models are the ratios of the variables derived from Internet consumption, such as the amount of consumed megabytes, and the frequency of use. It means that community detection improves the value to the classification process and therefore to work.

In summary, based on the presented metrics, the best model for classifying competitor carriers subscribers in those who will leave their carrier to become subscribers of the carrier running our methodology from those who would not is *Extremely Random Trees classifier*. This classification algorithm applied to the *total* dataset balanced with Random under-sampling technique out performs the other configurations. This result makes sense because this is the dataset

Proofreading version

that contains the all variables and therefore better identifies the behavior of the subscribers. It is worth noting that the difference in training the Extremely Random Trees classifier with the other datasets, such as calls, minutes or average after having applied the same resampling technique is not significantly different. Nevertheless, Ramdom Undersampling and Near-Miss 3 give slightly better results due to the majority points are near one to each other, which is the best scenario for both methods.

6 Conclusion

The construction of three graphs based on minutes, calls and average minutes per call as representations of the mobile telephone network allow us to analyze the interactions between subscribers. The value of this information relies especially on discovering the attributes and understanding the behavior of the competitors' customers. Based on these structures, important variables are generated for the classification of subscribers in those who will leave their carrier to become subscribers of the carrier running our methodology from those who would not. In addition, the community detection in the social graphs improve the characterization of subscribers from competitors carriers because we are able to infer certain consumption values (*e.g.*, consumed minutes, megabytes, among others) not visible in the dataset produced in the carrier running the proposed methodology. Also, it allows to obtain variables describing the behavior of the user in the community to which the subscriber belongs.

All the variables computed from subscriber interactions in the graph or in the community come together to characterize and obtain a refined individual subscriber profile. In addition, the time dimension is added, that is, how the attributes of the users vary between periods S1, S2, S3 and S4. Using these values, a complete characterization of the competitors subscribers behavior is obtained using. Therefore, this characterization takes into account the period as well as the evolution of these variables from one period to another. As observed in the results, the most significant variables for the best classifiers are the internet volume ratios and times (days) of consumption.

The selected Extremely Random Trees because classification algorithm uses bagging techniques of attributes and elements to perform the binary classification with interesting results. For the classification, the dataset from the first three periods S1, S2, and S3 are used to train the classifier. Subsequently, the performance of the model is measured with the fourth period S4. Note that due to the imbalance in the data, different resampling techniques are tested to balance classes and not bias the classifier by the majority class. Finally, based on the performance of the classifier after the application of a resampling technique, a comparative analysis is done. It is worth noting that our method could be applied to other industries when it is possible to model the clients interaction with competitors clients in a graph form

The classifier evaluation attained good values for ROC curve, AUC, Lift curve, and Lift value. However, due to the imbalanced nature of the dataset,

Proof Version

it is expected that not all the metrics reach desired values as shown in the PCR curve. Therefore, a low value in precision implies that not all subscribers identified as new potential subscribers will leave their current carrier. This opens new research opportunities to improve precision using deep learning and fuzzy logic techniques.

References

1. Amin, A., Shah, B., Khattak, A.M., Baker, T., Anwar, S., et al.: Just-in-time customer churn prediction: With and without data transformation. In: 2018 IEEE Congress on Evolutionary Computation (CEC). pp. 1–6. IEEE (2018)
2. Columelli, L., Nunez-del Prado, M., Zarate-Gamarra, L.: Measuring churner influence on pre-paid subscribers using fuzzy logic. In: 2016 XLII Latin American Computing Conference (CLEI). pp. 1–10. IEEE (2016)
3. De Caigny, A., Coussement, K., De Bock, K.W.: A new hybrid classification algorithm for customer churn prediction based on logistic regression and decision trees. European Journal of Operational Research **269**(2), 760–772 (2018)
4. Dugué, N., Perez, A.: Directed Louvain: maximizing modularity in directed networks. Ph.D. thesis, Université d'Orléans (2015)
5. Fawcett, T.: An introduction to roc analysis. Pattern Recognit Letters p. 861874 (2006)
6. Leicht, E.A., Newman, M.E.: Community structure in directed networks. Physical review letters **100**(11), 118703 (2008)
7. Lemaître, G., Nogueira, F., K.Aridas, C.: Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. Journal of Machine Learning Research **18**(17), 1–5 (2017), http://jmlr.org/papers/v18/16-365.html
8. del Prado, M.N., Hendrikx, H.: Toward a Route Detection Method base on Detail Call Records. Working Papers 16-19, Centro de Investigacin, Universidad del Pacfico (Dec 2016), https://ideas.repec.org/p/pai/wpaper/16-19.html
9. Saito, T., M., R.: Precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. PLoS ONE **10**(3), 1–21 (2015)
10. Shobha, G., et al.: Social network classifier for churn prediction in telecom data. In: 2013 International Conference on Advanced Computing and Communication Systems. pp. 1–7. IEEE (2013)
11. Wu, X., Liu, Z.: How community structure influences epidemic spread in social networks. Physica A: Statistical Mechanics and its Applications **387**(2-3), 623–630 (2008)
12. Zhu, M.: Recall, precision and average precision. University of Waterloo (2004)

Pre-publication version